

AMENDMENTS TO THE CLAIMS

Claims 2-8, 10, 12, 14-16, 19-21, 23-25, 29-41, 43, and 45-49 are currently pending in the Application. Claims 19-21, 23-25, 40, and 48 are allowable. Pursuant to the telephonic discussion between the Examiner of record and the undersigned on February 04, 2008, Applicants respectfully amend the specification and claims 21, 23, 40-41, and 48-49. A complete listing of claims is provided below and supersedes all previous listing(s) of claims. No new matter has been added.

1. (Canceled)
2. (Previously Presented) The method of claim 12, wherein the building the new stream comprises adding data to the operation code.
3. (Original) The method of claim 2, wherein the data comprises one or more values representative of a length of an instruction.
4. (Original) The method of claim 2, wherein the adding data to the operation code comprises adding a first variable before an instruction data of the content stream and a second variable after an instruction data of the content stream.
5. (Previously Presented) The method of claim 12, wherein the building the new stream comprises removing the operand from the content stream.
6. (Original) The method of claim 5, wherein the content stream comprises one or more additional operation code, and the method further comprising sorting the operation codes after the operand is removed.
7. (Original) The method of claim 5, wherein the building the new stream further comprises removing a noise operation code from the content stream.

8. (Previously Presented) A method for processing network traffic content, the method comprising:

receiving a content stream having an operation code and an operand;

building a new stream by either adding data to the operation code or removing the

operand from the content stream, wherein the building the new stream comprises

removing the operand from the content stream, and removing a noise operation code

from the content stream; and

storing the new stream in a computer readable medium;

wherein the content stream comprises one or more additional operation code, and the

method further comprising sorting the operation codes of the content stream after the

noise operation code is removed.

9. (Canceled)

10. (Previously Presented) The method of claim 12, wherein the model is selected from the group consisting of a Strict model, a Normal model, a Free model, and a Quiet model.

11. (Canceled)

12. (Previously Presented) A method for processing network traffic content, the method comprising:

receiving a content stream having an operation code and an operand;

building a new stream by either adding data to the operation code or removing the

operand from the content stream;

storing the new stream in a computer readable medium; and

creating a model using at least a portion of the new stream, wherein the creating comprises performing a cyclic redundancy check on at least a portion of the new stream, and wherein the at least a portion of the new stream comprises a first 12 bytes, a first 18 bytes, a first 24 bytes, a first 30 bytes, or a first 36 bytes, of the new stream.

13. (Canceled)

14. (Previously Presented) A method for processing network traffic content, the method comprising:

receiving a content stream having an operation code and an operand;
building a new stream by either adding data to the operation code or removing the operand from the content stream;
storing the new stream in a computer readable medium; and
creating a model using at least a portion of the new stream;
providing a bit table for the model;
searching the bit table to determine if the model is represented by a bit set in the bit table;
providing a cyclic redundancy check table for the model, the cyclic redundancy check table having a plurality of cyclic redundancy check elements; and
searching the cyclic redundancy check table to determine if the model matches with one of the plurality of cyclic redundancy check elements in the cyclic redundancy check table.

15. (Original) The method of claim 14, further comprising verifying a detection of a content when the model matches with a cyclic redundancy check element in the cyclic redundancy check table.

16. (Original) The method of claim 15, wherein the verifying comprises comparing a parameter obtained during a processing of the content stream with a verifier of a cyclic redundancy check element.

17-18. (Canceled)

19. (Previously Presented) The system of claim 23, wherein the model is selected from the group consisting of a Strict model, a Normal model, a Free model, and a Quiet model.

20. (Previously Presented) The system of claim 23, wherein the means for creating performs a cyclic redundancy check on at least a portion of the new stream.

21. (Currently Amended) A system for processing network traffic content, the system comprising:

means for receiving a content stream having an operation code and an operand;

means for building a new stream by either adding data to the operation code or removing the operand from the content stream; and

means for creating a model using at least a portion of the new stream, wherein the means for creating performs a cyclic redundancy check on at least a portion of the new stream,[[;]]

wherein the means for creating a model comprises a processor, and

wherein the at least a portion of the new stream comprises a first 12 bytes, a first 18 bytes, a first 24 bytes, a first 30 bytes, or a first 36 bytes, of the new stream.

22. (Canceled)

23. (Currently Amended) A system for processing network traffic content, the system comprising:

means for receiving a content stream having an operation code and an operand;

means for building a new stream by either adding data to the operation code or removing the operand from the content stream;

means for creating a model using at least a portion of the new stream;

means for providing a bit table for the model;

means for searching the bit table to determine if the model is represented by a bit set in the bit table; and

means for providing a cyclic redundancy check table for the model, the cyclic redundancy check table having a plurality of cyclic redundancy check elements; and

means for searching the cyclic redundancy check table to determine if the model matches with one of the plurality of cyclic redundancy check elements in the cyclic redundancy check table, wherein the means for searching the cyclic redundancy check table comprises a processor.

24. (Original) The system of claim 23, further comprising means for verifying a detection of a content when the model matches with a cyclic redundancy check element in the cyclic redundancy check table.

25. (Original) The system of claim 24, wherein the means for verifying compares a parameter obtained during a processing of the content stream with a verifier of a cyclic redundancy check element.

26-28. (Canceled)

29. (Previously Presented) A method for processing network traffic content, the method comprising:

receiving a content stream, the content stream having an operation code and an operand;
building a new stream by either adding data to the operation code or removing the
operand from the content stream;
creating a first model using at least a portion of the new stream;
storing the first model in a computer readable medium; and
searching a first cyclic redundancy check table to determine if the first model matches
with a cyclic redundancy check element stored in the first cyclic redundancy check
table.

30. (Previously Presented) The method of claim 29, wherein the building the new stream
comprises adding data to the operation code.

31. (Previously Presented) The method of claim 30, wherein the adding data to the operation
code comprises adding a first variable before an instruction data of the content stream and a
second variable after an instruction data of the content stream.

32. (Original) The method of claim 29, further comprising:
creating a second model using raw data of the content stream; and
searching a second cyclic redundancy check table to determine if the second model
matches with a cyclic redundancy check element stored in the second cyclic
redundancy check table.

33. (Original) The method of claim 32, wherein the creating the second model comprises
removing the operand from the content stream.

34. (Original) The method of claim 32, further comprising:
creating a third model using raw data of the content stream; and

searching a third cyclic redundancy check table to determine if the third model matches with a cyclic redundancy check element stored in the third cyclic redundancy check table.

35. (Original) The method of claim 34, wherein the content stream comprises one or more additional operation codes, and wherein the creating the third model comprises:

removing the operand from the content stream; and
sorting the operation codes of the content stream after the removing.

36. (Original) The method of claim 29, further comprising:
creating a quiet model using raw data of the content stream; and
searching a quiet bit table to determine if the quiet model is represented by a bit set stored in the quiet bit table, wherein the searching the quiet bit table is performed before the searching the first cyclic redundancy check table.

37. (Original) The method of claim 36, further comprising:
creating a strict model using raw data of the content stream; and
searching a strict bit table to determine if the strict model is represented by a bit set stored in the strict bit table.

38. (Original) The method of claim 37, further comprising:
creating a normal model using raw data of the content stream; and
searching a normal bit table to determine if the normal model is represented by a bit set stored in the normal bit table.

39. (Previously Presented) The method of claim 38, further comprising:
creating a free model using raw data of the content stream; and

searching a free bit table to determine if the free model is represented by a bit set stored in the free bit table.

40. (Currently Amended) A system for processing network traffic content, the system comprising:

means for receiving a content stream, the content stream having an operation code and an operand;

means for creating a new stream by either adding data to the operation code or removing the operand from the content stream;

means for creating a model using at least a portion of the new stream; and

means for searching a cyclic redundancy check table to determine if the model matches with a cyclic redundancy check element stored in the cyclic redundancy check table, wherein the means for searching a cyclic redundancy check table comprises a processor.

41. (Currently Amended) A ~~computer~~ product that includes a tangible readable medium usable by a processor and has ~~having~~ a set of stored instructions, the execution of which causes a the processor to execute a process for processing network traffic content ~~to be performed~~, the process comprising:

receiving a content stream, the content stream having an operation code and an operand;

creating a new stream by either adding data to the operation code or removing the operand from the content stream;

creating a model using at least a portion of the new stream;

storing the first model in a computer readable medium; and

searching a cyclic redundancy check table to determine if the model matches with a cyclic redundancy check element, wherein the cyclic redundancy check element is stored in the cyclic redundancy check table in a tangible volatile or non-volatile medium.

42. (Canceled)

43. (Previously Presented) The method of claim 46, wherein the first model comprises a quiet model.

44. (Canceled)

45. (Previously Presented) The method of claim 46, wherein the second model is selected from the group consisting of a strict model, a normal model, and a free model.

46. (Previously Presented) A method for processing network traffic content, the method comprising:

receiving a content stream;

creating a first model using raw data of the content stream;

storing the first model in a computer readable medium;

searching a first bit table to determine if the first model is represented by a bit set stored in the first bit table;

creating a second model using raw data of the content stream;

searching a second bit table to determine if the second model is represented by a bit set stored in the second bit table; and

searching a cyclic redundancy check table to determine if the second model matches with a cyclic redundancy check element stored in the cyclic redundancy check table.

47. (Original) The method of claim 46, wherein the cyclic redundancy check table is selected from the group consisting of a strict cyclic redundancy check table, a normal cyclic redundancy check table, and a free cyclic redundancy check table.

48. (Currently Amended) A system for processing network traffic content, the system comprising:

means for receiving a content stream;

means for creating a first model using raw data of the content stream;

means for searching a first bit table to determine if the first model is represented by a bit set stored in the first bit table;

means for creating a second model using raw data of the content stream;

means for searching a second bit table to determine if the second model is represented by a bit set stored in the second bit table; and

means for searching a cyclic redundancy check table to determine if the second model matches with a cyclic redundancy check element stored in the cyclic redundancy check table, wherein the means for searching a cyclic redundancy check table comprises a processor.

49. (Currently Amended) A computer product that includes a tangible readable medium usable by a processor ~~having a computer readable medium that includes~~ having a set of stored instructions, the execution of which causes ~~[[a]] the processor to execute a process for processing network traffic content to be performed,~~ the process comprising:

receiving a content stream;

creating a first model using raw data of the content stream;

storing the first model; and

searching a first bit table to determine if the first model is represented by a bit set stored in the first bit table;

creating a second model using raw data of the content stream;

searching a second bit table to determine if the second model is represented by a bit set stored in the second bit table; and

searching a cyclic redundancy check table to determine if the second model matches with a cyclic redundancy check element, wherein the cyclic redundancy check element is stored in the cyclic redundancy check table in a tangible volatile or non-volatile medium.